Initialisation Determines the Basin: Efficient Codebook Optimisation for Extreme LLM Quantization

Ian W. Kennedy & Nafise Sadat Moosavi
School of Computer Science
University of Sheffield
Sheffield, UK
{iwkennedy1,n.s.moosavi}@sheffield.ac.uk

Abstract

Additive quantization enables extreme LLM compression while preserving lookup-table inference with free-form codebooks. At 2 bits per weight, however, greedy sequential initialisation can cause large quality losses that wider assignment search and PV-tuning do not fully erase. We show that this failure reflects an interaction between representational pressure and sequential codebook coupling. We use $\rho = N/K^M$ as a structural pressure measure and propose OA-EM, a hard generalised-EM initialisation procedure using Hessian-weighted Mahalanobis distance to minimise output-aware reconstruction error. Across four models from 3B to 14B parameters, OA-EM consistently lowers matched post-PV perplexity. Controlled experiments at a fixed bitrate show that ρ alone does not explain sensitivity to initialisation; the number and size of the additive codebooks also matter. A Euclidean-EM ablation and curvature diagnostics show that the gain is not explained by EM refinement alone: OA-EM reduces error in high-curvature directions. On Llama 3.2 3B, OA-EM also improves the measured quality-compute trade-off. Overall, the final quantized solution remains persistently dependent on initial codebook geometry. Code: https://github.com/kenno94-IK/aqlm-oaem.

1 Introduction

Large language model deployment on consumer hardware requires aggressive weight compression (Gholami et al., 2021). While 4-bit quantization is near-lossless (Frantar et al., 2023; Lin et al., 2024), the 2-bit-per-weight regime remains challenging because the representation budget is extremely constrained. For vector and additive quantizers, "2-bit" denotes the average bitrate rather than four scalar values per parameter. This operating point is relevant whenever model memory is the deployment bottleneck, including consumer-GPU and edge settings.

Two paradigms compete in this regime. *Structured* methods—such as lattice codebooks (Tseng et al., 2024a), trellis codes (Tseng et al., 2024b), and grouped lattice VQ (Zhang et al., 2025)—achieve strong perplexity via constrained codebook geometry and specialised decoding procedures. In contrast, *free-form* additive methods (Egiazarian et al., 2024) learn unconstrained codebooks and retain AQLM's lookup-table dequantization pathway. Our contribution concerns optimisation quality within this free-form family; we do not claim runtime superiority over structured methods without matched deployment benchmarks.

Additive quantization (Egiazarian et al., 2024) encodes each group of g weights as a sum of M codewords, drawn from learned codebooks of $K{=}256$ entries each. When performance degrades at extreme compression, the common response is to increase computational effort: wider beam search, more epochs per layer, or larger calibration sets. We show that, in this regime, such strategies target the wrong bottleneck. A major factor is initialisation: search performed within a poorly initialised region of the solution space yields limited improvement. We use the ratio of weight groups to nominal codeword tuples as a structural descriptor of one source of pressure:

Definition 1 (Representational Ratio)**.** *For M additive codebooks with K entries each and N weight groups in a quantized weight matrix, the representational ratio is $\rho = N/K^M$. The ratio is structural: it is determined by the matrix dimensions and quantizer configuration (N, K, M) and therefore does not change after OA-EM or PV-tuning. Unless otherwise stated, scalar ρ values in the main text use a representative matrix; Appendix E reports the matrix-level analysis across model scales and explains why we do not treat ρ as a sufficient predictor.*

We interpret ρ as a measure of representational

Proceedings of the 15th International Joint Conference on Natural Language Processing and the 5th Conference of the Asia-Pacific Chapter of the Association for Computational Linguistics, pages 1966–1981
November 6-10, 2026 ©2026 Association for Computational Linguistics

pressure, not as a sufficient predictor of failure. Low ρ provides many representable combinations per observed group, whereas high ρ increases competition for codebook capacity. Catastrophic sensitivity can be amplified by sequential coupling: when $M > 1$, an error in an early residual codebook changes the residual distribution presented to later codebooks. Fixed-bitrate experiments in §5.2 probe these effects, although the decompositions change multiple structural properties. Table 1 gives the two standard operating points for Llama 3.2 3B.

Rate	M	K^M	ρ	Regime
3 bpp	3	16.8M	≈ 0.07	Overcomplete
2 bpp	2	65K	≈ 18	Undercomplete

Table 1: Representational ratio ρ for a representative Llama 3.2 3B matrix (K=256, g=8).

The change in representational pressure coincides with a large increase in sensitivity for the standard residual configuration, although our controlled experiments show that ρ is not sufficient on its own. At 3 bpp, greedy initialisation incurs a 0.65-point penalty in perplexity (Table 4). At 2 bpp, it leads to severe degradation on Llama 3.2 3B (WikiText-2 perplexity 352.39 at beam 4, 60.61 at beam 8 vs. 7.28 FP16), and even quadrupling the beam width to 16 reduces this only to 46.01 (Table 2).

To address this, we propose **OA-EM** (Output-Aware Expectation-Maximisation), which refines each codebook's initialisation via iterative EM (Dempster et al., 1977) using Hessian-weighted Mahalanobis distance derived from calibration activations, minimising a curvature-weighted approximation of output reconstruction error rather than weight-space distance.

Beyond the algorithm itself, our primary contribution is the empirical and analytical observation that *initialisation strongly influences the optimisation trajectory of compressed models*. The experiments support considering both structural pressure and the additive decomposition: ρ describes nominal tuple availability, while the depth of the sequential fitting chain determines how far early placement errors propagate. The experiments do not isolate chain depth from all other geometric changes. We demonstrate this through multiple lines of evidence:

Basin persistence. OA-EM's advantage persists after end-to-end PV-tuning (Malinovskii et al., 2024) across beam widths, epoch budgets, compression rates, model scales, and model families. On Llama 3.2 3B at b=8, PV-tuning reduces a 43-point perplexity gap to 0.23, yet OA-EM remains better across all configurations (Table 3).

Controlled mechanism tests. Fixed-bitrate decompositions show that high ρ alone does not guarantee catastrophic failure, while Euclidean-EM and curvature ablations isolate the benefit of output-aware refinement beyond generic EM (§5.2, Appendix D).

Quality–compute trade-off. On Llama 3.2 3B, OA-EM improves the measured quality–compute trade-off in the A100 sweep: at each full-search beam width, it achieves lower post-PV perplexity and higher average downstream accuracy with equal or lower quantization time (Table 5).

Contributions.

1. We introduce ρ as a structural measure of representational pressure and use controlled fixed-bitrate decompositions to show how it interacts with sequential codebook coupling (§3.3, §5.2).

2. We propose OA-EM—output-aware EM with Hessian-weighted Mahalanobis distance—as a drop-in replacement for greedy initialisation, and isolate its mechanism against Euclidean EM and curvature-aware diagnostics (§3.4).

3. Through post-PV analysis across beam widths, compression rates (2 bpp and 3 bpp), and four models up to 14B parameters (Llama 3.2 3B, Llama 3.1 8B, Qwen 2.5 3B, Qwen 2.5 14B), we show persistent dependence of final perplexity on initial codebook geometry (§5.3, §6).

2 Related Work

Scalar PTQ. A large body of work studies scalar post-training quantization (PTQ) for LLM compression, where each weight is assigned an independent low-bit representation. GPTQ (Frantar et al., 2023) quantizes layer-wise using approximate second-order information; AWQ (Lin et al., 2024) protects salient weights via activation-based scaling. SmoothQuant (Xiao et al., 2023) migrates quantization difficulty from activations to weights via per-channel scaling, and OmniQuant (Shao et al., 2024) learns weight clipping and equivalence transformations end-to-end.

Such scalar methods degrade rapidly below 3 bpp because scalar codes cannot efficiently use the information budget at extreme compression. LLM.int8() (Dettmers et al., 2022) demonstrated mixed-precision decomposition for 8-bit quantization. SpQR (Dettmers et al., 2024) and SqueezeLLM (Kim et al., 2024) extend sensitivity-aware principles to 3–4 bits. QLoRA (Dettmers et al., 2023) combines 4-bit quantization with low-rank adaptation for efficient fine-tuning, demonstrating the practical demand for aggressive compression on consumer hardware. These scalar approaches share our motivation that weight sensitivity should guide compression, but operate in the regime where the initialisation problem we study does not arise.

Structured vector quantization for LLMs. Beyond scalar quantization, several works explore vector quantization with structured codebooks for extreme compression. QuIP (Chee et al., 2023) enabled 2-bit LLMs via incoherence processing. QuIP# (Tseng et al., 2024a) introduced the randomised Hadamard transform and E8 lattice codebooks. Because the codebook structure is fixed rather than learned, QuIP# avoids the initialisation problem we study. SpinQuant (Liu et al., 2025) learns rotation matrices to remove outliers and improve quantization accuracy; like QuIP#, it transforms the weight distribution rather than learning codebooks, so the initialisation bottleneck does not apply. QTIP (Tseng et al., 2024b) achieves strong 2-bit results via trellis-coded quantization.

Grouped lattice VQ. A recent line of work further develops structured codebooks through lattice-based constructions. GLVQ (Zhang et al., 2025) assigns each weight group a customised lattice codebook defined by a learnable generation matrix, achieving strong 2-bit perplexity by combining structured codebook geometry with per-group adaptability. GLVQ sidesteps the free-form initialisation problem studied here by imposing lattice structure and using lattice decoding. Its inference procedure therefore differs from AQLM's learned free-form lookup-table pathway. A matched latency comparison is outside the scope of this work, so we treat the methods as addressing different design points rather than claiming runtime superiority. Our work is orthogonal to GLVQ: we study whether free-form additive codebooks can avoid poor optimisation trajectories while retaining the existing AQLM inference representation.

Additive quantization for LLMs. Additive quantization represents weights as sums of codewords from multiple codebooks, allowing a much larger set of representable values than scalar quantization at the same bitrate. AQLM (Egiazarian et al., 2024) adapts multi-codebook quantization from information retrieval (Babenko and Lempitsky, 2014; Jégou et al., 2011) to LLM compression. PV-tuning (Malinovskii et al., 2024) extends AQLM with end-to-end optimisation of continuous codebook parameters and discrete code assignments. Neither work examines how initialisation quality interacts with compression rate or persists through fine-tuning, which is the focus of our paper. This question is particularly important at extreme compression, where poor initialisation can cause additive quantization to fail despite sufficient representational capacity.

EM-based VQ for LLMs. The Expectation-Maximisation (EM) algorithm has previously been applied to improve vector quantization. GPTVQ (van Baalen et al., 2025) applies EM-based VQ within the GPTQ framework for LLM compression, using a single learned codebook and performing EM directly for quantization. LSQ++ (Martinez et al., 2018) similarly applies EM to additive quantization in information retrieval. Our setting differs in two important respects. First, OA-EM operates within additive quantization for LLMs, where multiple residual codebooks interact combinatorially and early-codebook errors alter the residuals presented to later codebooks. Second, OA-EM is output-aware, optimising reconstruction error in activation space and serving specifically as an initialisation stage for downstream beam search rather than as the primary quantizer. The Hessian-weighted distance itself is not our novelty. The contribution is in identifying sequential additive-codebook coupling as the failure mode, formalising the representational pressure under which it becomes important, and using output-aware EM in this initialisation role. Single-codebook GPTVQ does not contain the same residual-chain coupling.

Positioning. Existing work improves different stages of the vector quantization pipeline. Structured methods (QuIP#, QTIP, GLVQ) improve *quantizer design* through codebook geometry; GPTVQ improves the *codebook learning algorithm*; and PV-tuning improves *post-quantization fine-tuning*. Our work is orthogonal: we improve the *initialisation stage* and show that codebook ini-

tialisation determines the persistent optimisation basin that subsequent search and fine-tuning reach. Rather than proposing a new quantizer geometry, we show that initialisation quality is a previously overlooked bottleneck that can dominate performance at extreme compression within free-form additive quantization. The mechanism we study is most directly relevant to learned-codebook quantizers that use sequential residual fitting.

3 Method

We first review the additive quantization framework used in AQLM, then analyse why greedy initialisation becomes a bottleneck at extreme compression. Finally, we introduce OA-EM, an output-aware EM algorithm that improves codebook initialisation and yields better optimisation basins for downstream search and fine-tuning.

3.1 Background: AQLM

Additive quantization for LLMs is implemented in AQLM, which represents each weight group $\mathbf{w} \in \mathbb{R}^g$ as a sum of M codewords, $\hat{\mathbf{w}} = \sum_{m=1}^{M} \mathbf{c}_{m,b_m}$, where $\mathbf{c}_{m,b_m}$ is the b_m-th entry of codebook $\mathcal{C}_m \in \mathbb{R}^{K\times g}$, with $K{=}256$. At 2 bpp with group size $g{=}8$, AQLM uses $M{=}2$ codebooks: each group of 8 weights is represented by two 8-bit indices.

The layer-wise objective minimises output reconstruction error on calibration data, recently shown to be linearly predictive of model perplexity increase (Malinovskii et al., 2025):

$$\mathcal{L} = \|\mathbf{X}\mathbf{W} - \mathbf{X}\hat{\mathbf{W}}\|_F^2 \tag{1}$$

where $\mathbf{X}$ is the calibration activation matrix. Optimisation proceeds in two stages. Codebooks are first initialised via residual k-means, after which assignments are refined by beam search over the combinatorial space of codeword combinations.

3.2 Beam Search: The Standard Remedy

Greedy sequential assignment—selecting the best entry from each codebook in turn without revisiting earlier choices—suffers from *premature commitment*: the best first-codebook entry in isolation may pair poorly with available second-codebook entries, but greedy search discards all alternatives, making such errors irrecoverable. AQLM's standard remedy is beam search with width b, which defers commitment by maintaining b active candidates at each codebook stage, bridging the gap between greedy search ($b{=}1$) and exhaustive enumeration ($b{=}K^{M-1}$) at a cost of $O(MbK)$ per weight group.

However, wider beams are expensive: on Llama 3.2 3B at 2 bpp, increasing b from 4 to 16 raises quantization time from 6.1h to 16.9h, a $2.8\times$ cost for a reduction from 352.39 to only 46.01 on WikiText-2 perplexity (Table 2). More importantly, beam search optimises *assignments* over a fixed set of codebook centroids. If the centroids themselves are poorly placed, even the globally optimal assignment will yield high reconstruction error—beam search finds the best path through a bad tree, but cannot reshape the tree itself. This reveals the deeper limitation we address: the bottleneck is not insufficient search over assignments, but poor codebook geometry. OA-EM (§3.4) addresses the root cause directly, improving centroid placement so that even narrow-beam search operates in a favourable landscape.

3.3 The Initialisation Bottleneck

Residual k-means fits codebooks greedily: $\mathcal{C}_1$ is fitted to the weight vectors, then $\mathcal{C}_2$ is fitted to the residuals $\mathbf{w} - \mathbf{c}_{1,b_1}$. This sequential procedure ignores the joint structure; the optimal $\mathcal{C}_1$ depends on what $\mathcal{C}_2$ can represent, and vice versa. We now analyse how this coupling leads to suboptimal assignments and why the effect becomes severe when representational capacity is limited ($\rho > 1$).

Proposition 1 (Greedy Suboptimality Decomposition)**.** *Let (i^*, j^*) denote the optimal assignment for weight group $\mathbf{w}$, and (i^g, j^g) the greedy sequential assignment. Let $\boldsymbol{\delta} = \mathbf{c}_{1,i^g} - \mathbf{c}_{1,i^*}$ be the first-codebook displacement. The suboptimality gap is:*

$$\varepsilon^g - \varepsilon^* = \underbrace{\|\boldsymbol{\delta}\|^2}_{\textit{direct cost}} + \underbrace{2\langle \boldsymbol{\delta},\, \mathbf{c}_{2,j^g} - \mathbf{r}^* \rangle}_{\textit{coupling}} + \underbrace{\|\mathbf{r}^* - \mathbf{c}_{2,j^g}\|^2 - \varepsilon^*}_{\textit{residual mismatch} \geq 0} \tag{2}$$

where $\mathbf{r}^* = \mathbf{w} - \mathbf{c}_{1,i^*}$ is the joint-optimal residual.

Proof. See Appendix I.

The decomposition reveals three sources of greedy error. The *direct cost* $\|\boldsymbol{\delta}\|^2$ is the squared distance between greedy and optimal first-codebook entries. The *coupling term* captures how well $\mathcal{C}_2$ can compensate for $\boldsymbol{\delta}$—when $\mathcal{C}_2$ has an entry near $\mathbf{r}^* - \boldsymbol{\delta}$, this term can be negative. The *residual mismatch* is always non-negative for the fixed codebooks and assignments in Proposition 1.

Error correction capacity. After committing to codebook 1, the remaining codebooks provide K^{M-1} possible residual configurations. At 3 bpp (M=3), $K^2 = 65{,}536$ configurations allow the remaining codebooks to absorb most $\boldsymbol{\delta}$ displacements. At 2 bpp (M=2), only $K = 256$ entries are available, a 256× reduction in correction capacity. Consequently, the second codebook must simultaneously represent the true weight structure *and* compensate for errors in the first codebook, making greedy placement much more brittle. This combinatorial correction capacity is related to, but not fully described by, ρ. High ρ means many observed groups share a limited set of representable combinations; sequential residual fitting determines how errors propagate through those combinations. In particular, M=1 removes residual-chain coupling even when ρ is high, while increasing M creates more opportunities for early-codebook errors to reshape later residuals. The decompositions in §5.2 probe these factors, with group size, codebook size, or parameter count also changing. Structured methods such as GLVQ (Zhang et al., 2025) sidestep this problem entirely by replacing free-form codebooks with lattice geometry, eliminating the combinatorial assignment; OA-EM instead solves the problem within the free-form paradigm by improving the initial codebook geometry while preserving the LUT dequantization pathway.

3.4 OA-EM: Output-Aware EM Initialisation

OA-EM improves upon k-means initialisation by replacing Euclidean distance with Hessian-weighted Mahalanobis distance in both centroid optimisation and code assignment. OA-EM operates within AQLM's residual framework—codebooks are still fitted sequentially—but refines each codebook using hard generalised EM: the E-step uses hard assignments and the M-step approximately minimises the objective with a finite number of Adam updates. This procedure targets a block-diagonal approximation of the reconstruction objective (Eq. 1).

Starting from k-means-initialised centroids $\{\mathbf{c}_k\}_{k=1}^{K}$ and assignments $\{b_i\}_{i=1}^{N}$, OA-EM alternates two steps for R rounds:

M-step (centroid optimisation). Fix assignments and optimise centroids to minimise the Hessian-weighted reconstruction error:

$$\mathcal{L}_{\text{EM}} = \frac{1}{N}\sum_{i=1}^{N} \mathbf{e}_i^\top \mathbf{H}_i\, \mathbf{e}_i, \quad \mathbf{e}_i = \mathbf{w}_i - \mathbf{c}_{b_i} \tag{3}$$

where $\mathbf{H}_i = \mathbf{X}_i^\top \mathbf{X}_i + \lambda \mathbf{I}$ is the damped block-diagonal Hessian approximation for weight group i, following the use of second-order information for quantization in GPTQ (Frantar et al., 2023) and GPTVQ (van Baalen et al., 2025); the damping constant is $\lambda = 0.01 \cdot \overline{\text{diag}(\mathbf{H})}$.

Centroids are updated via S Adam steps with cosine learning rate annealing from η to 0.1η.

E-step (hard reassignment). Fix centroids and reassign each weight group to its nearest centroid under Mahalanobis distance:

$$b_i \leftarrow \arg\min_{k\in[K]} (\mathbf{w}_i - \mathbf{c}_k)^\top \mathbf{H}_i\, (\mathbf{w}_i - \mathbf{c}_k) \tag{4}$$

Connection to the greedy bound. OA-EM does not eliminate sequential fitting, but it aims to reduce the dominant error terms in Proposition 1. Euclidean k-means allocates first-codebook centroids largely according to weight magnitude: large-norm groups attract centroids regardless of their output sensitivity. When $\rho \gg 1$ and representational capacity is scarce, this wastes centroids on output-insensitive groups, producing a large displacement $\boldsymbol{\delta}$ for the output-sensitive groups that dominate the reconstruction loss. OA-EM's Hessian weighting reverses this allocation: the M-step concentrates centroids on groups with large $\mathbf{H}_i$, ensuring small $\boldsymbol{\delta}$ precisely where it matters most—directly reducing the $\|\boldsymbol{\delta}\|^2$ direct cost for groups that contribute most to Eq. 1. Improved first-codebook placement also reduces the residual mismatch term, because the residuals $\mathbf{r} = \mathbf{w} - \mathbf{c}_{1,b_1}$ passed to the second codebook more closely resemble the jointly optimal residuals $\mathbf{r}^*$. We use R=3 rounds of S=100 Adam steps with η=10^{-4}.

Mechanism diagnostic. A held-out 196-matrix analysis shows that OA-EM increases median weight NMSE by 4.55% while reducing leading-eigenspace curvature NMSE by 11.2–12.9%, winning 146–170 matrices. Total output NMSE is mixed, and `gate_proj` worsens at top-32. The evidence supports preferential preservation of high-curvature directions in aggregate; Appendix D.3 gives the full method and projection breakdowns.

4 Experimental Setup

Models. We evaluate four models: Llama 3.2 3B and Llama 3.1 8B (Grattafiori et al., 2024), and Qwen 2.5 3B and Qwen 2.5 14B (Yang et al., 2025), covering two model families and scales from 3B to 14B parameters. All models are quantized with AQLM at 2 bpp (K=256, M=2, g=8).

Llama 3.2 3B is additionally evaluated at 3 bpp (M=3, g=8), and in controlled 2-bpp decompositions that vary (K, M, g) to probe representational pressure and residual decomposition.

Calibration. We use 128 sequences from C4 (Raffel et al., 2020) with length 4096 for calibration, following common practice in post-training quantization (Frantar et al., 2023; Lin et al., 2024).

Beam search configurations. We vary beam width $b \in \{4, 8, 16\}$ and maximum epochs $e \in \{5, 100\}$ with early stopping at 0.01 relative MSE, spanning a 2.8× range in quantization time (6.1h to 16.9h on Llama 3.2 3B). Qwen 2.5 3B and 3 bpp are evaluated at b=8, e=100.

OA-EM configuration. OA-EM is run for 3 EM rounds with 100 Adam steps per round, learning rate $\eta = 10^{-4}$ with cosine annealing to 0.1η. Both E-step and M-step use the damped block-diagonal Hessian approximation $\mathbf{H}_i = \mathbf{X}_i^\top \mathbf{X}_i + \lambda \mathbf{I}$.

PV-tuning. We follow Malinovskii et al. (2024): Adam optimiser, learning rate $\eta = 3 \times 10^{-4}$, batch 32, 10K samples, five-epoch budget, and best-WikiText-2 checkpoint selection. Appendix C.4 gives the matched PV results for Qwen 2.5 14B.

Evaluation. We report perplexity on WikiText-2 (Merity et al., 2016) and C4 (4096 context). Zero-shot evaluation is performed on ARC-Easy and ARC-Challenge (Clark et al., 2018), HellaSwag (Zellers et al., 2019), PIQA (Bisk et al., 2020), WinoGrande (Sakaguchi et al., 2021), and LAMBADA (Paperno et al., 2016), using the LM Evaluation Harness (Sutawika et al., 2024).

Hardware and seeds. Llama 3.2 3B and Qwen 2.5 3B were quantized and PV-tuned on a single NVIDIA A100 80 GB. Llama 3.1 8B was quantized on a single A100 80 GB and PV-tuned on a single B200 192 GB. Qwen 2.5 14B was quantized on one B200 and PV-tuned on two. Unless otherwise stated, results use seed 42; Appendix F reports seed-0 and seed-7 runs.

5 Initialisation and Basin Persistence

We study how strongly the final quantized solution depends on codebook initialisation. We use "basin persistence" operationally to mean that matched optimisation from different initialisations yields persistently different final solutions; we do not claim to characterise the topology of the full non-convex loss landscape. Pre-PV results isolate the effect of initialisation on quantization optimisation, while post-PV results test whether that dependence remains after a strong downstream optimiser. We conduct this analysis on Llama 3.2 3B as a representative model across a wide range of beam-search configurations.

	Wiki-2	C4	Time
FP16	7.28	11.04	—
Greedy initialisation			
b=4, e=100	352.39	20.66	6.1h
b=8, e=100	60.61	18.64	9.9h
b=16, e=100	46.01	19.00	16.9h
b=8, e=5	85.72	47.04	6.8h
OA-EM initialisation			
b=4, e=100	16.82	18.09	6.1h
b=8, e=100	17.39	18.00	9.2h
b=16, e=100	**16.53**	**17.92**	15.5h
b=8, e=5	18.91	17.98	7.3h

Table 2: Pre-PV-tuning results at 2 bpp on Llama 3.2 3B. Beam width b, maximum epochs e. Quantization time on one A100.

5.1 Initialisation Effects Before PV-Tuning

Overcomplete Regime (3 bpp, $\rho \approx 0.07$). At 3 bpp, the initialisation bottleneck is relatively mild. OA-EM reduces WikiText-2 perplexity from 9.52 to 8.87 (−0.65), while C4 increases slightly from 13.39 to 13.51. OA-EM reduces quantization time by 5.7% (12h 39m vs. 13h 25m), as improved initialisation requires fewer beam-search epochs per layer. Full pre-PV-tuning results are in Appendix B.

Undercomplete Regime (2 bpp, $\rho \approx 18$). The 2 bpp regime reveals the full impact of the initialisation bottleneck. Table 2 presents pre-PV-tuning results across beam-search configurations.

Three observations emerge. First, beam search alone cannot compensate for poor initialisation. For greedy initialisation, performance improves with wider beams (b=4: 352.39, b=8: 60.61, b=16: 46.01), but remains far from the OA-EM results. In contrast, OA-EM remains stable across beam widths (16.53–17.39).

Second, increasing the search budget does not consistently improve the baseline solution quality. While WikiText-2 perplexity improves with larger beams, C4 perplexity slightly worsens (18.64→19.00), showing that larger search budgets do not uniformly improve held-out perplexity. Third, OA-EM improves both quality and efficiency. For example, with b=8, OA-EM achieves

17.39/18.00 in 9.2h compared to 60.61/18.64 in 9.9h for greedy initialisation.

However, pre-PV-tuning results alone do not determine the practical impact of improved initialisation. PV-tuning can substantially improve quantized models, potentially compensating for poor initialisation. We therefore examine whether OA-EM's advantage persists after PV-tuning in §5.3.

5.2 Controlled Mechanism Checks

At fixed 2 bpp, changing the residual decomposition gives Greedy/OA-EM WikiText-2 perplexity of 21.39/15.66 for $(g{=}4, M{=}1)$, 60.61/17.39 for $(g{=}8, M{=}2)$, and 19.08/16.94 for $(g{=}16, M{=}4)$. The markedly different Greedy–OA-EM gaps show that ρ alone is insufficient: residual-chain structure materially affects sensitivity. Holding $K^M{=}65{,}536$ fixed, $(K{=}16, M{=}4)$ retains a 16.5% post-PV WikiText-2 advantage for OA-EM (17.16→14.33) despite near-complete codebook utilisation.

Euclidean EM improves pre-PV WikiText-2 over Greedy (60.61→43.41) but remains far behind OA-EM (17.39), and after PV-tuning it is worse than both OA-EM and Greedy (WikiText-2 12.20 vs. 11.53 and 11.76). Generic EM refinement therefore does not explain the gain; together with the curvature diagnostic (§3.4), this isolates the benefit of output-aware weighting. Full decompositions, C4 results, and ablation tables are reported in Appendix D.

5.3 Basin Persistence After PV-Tuning

PV-tuning (Malinovskii et al., 2024) performs end-to-end optimisation of continuous codebook parameters and discrete code assignments. Because PV-tuning substantially improves quantized models, it acts as a strong optimiser that could potentially erase differences caused by initialisation. A key question is therefore whether improved initialisation still leads to better final models, or whether PV-tuning causes all configurations to converge to the same solution. Table 3 reports WikiText-2 perplexity before and after PV-tuning across all quantization configurations on Llama 3.2 3B. PV-tuning substantially improves all models, dramatically reducing the gap caused by poor initialisation. However, OA-EM consistently achieves a lower final perplexity across all matched configurations. Thus, although PV-tuning mitigates poor initialisation, it does not eliminate dependence on the starting codebook geometry under this optimisation procedure.

Init	Pre-PV	Post-PV	Δ	Q-time
Narrow beam ($b{=}4$, $e{=}100$)				
Greedy	352.39	12.66	−339.73	6.1h
OA-EM	16.82	**11.53**	−5.29	6.1h
Standard beam ($b{=}8$, $e{=}100$)				
Greedy	60.61	11.76	−48.85	9.9h
OA-EM	17.39	**11.53**	−5.86	9.2h
Wide beam ($b{=}16$, $e{=}100$)				
Greedy	46.01	12.01	−34.00	16.9h
OA-EM	16.53	**11.49**	−5.04	15.5h
Early stopping ($b{=}8$, $e{=}5$)				
Greedy	85.72	12.69	−73.03	6.8h
OA-EM	18.91	**11.76**	−7.15	7.3h

Table 3: WikiText-2 perplexity before and after PV-tuning at 2 bpp on Llama 3.2 3B (seed 42). Q-time is quantization time on one A100 and excludes the ∼2.0h PV-tuning cost common to both initialisations (Appendix G).

Beam-width robustness. The fully matched seed-0 sweep shows that OA-EM improves both perplexity measures at every beam, before and after PV-tuning. An OA-EM seed-7 replication and the original seed-42 results show that the best beam depends on the seed and the dataset. We therefore interpret these experiments as evidence of robustness to search budget rather than monotonic improvement; Appendix F gives the numerical results and coverage limits.

Pareto improvements. In Table 3, OA-EM improves the quality–time trade-off. Within every configuration, OA-EM reaches lower post-PV perplexity with equal or lower quantization time, except the short-budget setting ($b{=}8$, $e{=}5$), where the fixed cost of OA-EM initialisation is not fully amortised. Across beam widths, OA-EM at $b{=}8$ (9.2h, 11.53) outperforms Greedy at $b{=}16$ (16.9h, 12.01) with 46% less quantization time. A full Pareto analysis is provided in Appendix A. PV-tuning adds approximately 2.0h on a single A100 for the 3B model with the same configuration across initialisations; Appendix G reports end-to-end wall-clock totals.

Downstream accuracy. OA-EM improves mean zero-shot accuracy in every evaluated Llama setting, by up to +1.6 points on Llama 3.2 3B (at $b{=}4$) and by +0.7 points on Llama 3.1 8B, while Qwen 2.5 3B shows a small regression (0.606→0.603) despite lower perplexity. We therefore treat perplexity as the primary matched signal and report the full task-level results and summary table in Appendix C.

6 Generality Across Compression Rates and Architectures

Table 4 summarises perplexity results across compression rates and architectures.

Representational pressure and sequential coupling. Across rates, lower representational pressure is sometimes associated with smaller Greedy–OA-EM gaps, but the controlled decompositions in §5.2 show that ρ is not sufficient by itself. A per-matrix analysis across the evaluated 3B, 8B, and 14B scales reaches the same conclusion: overall correlations of $\log_{10}\rho$ with relative reconstruction-error reduction are weakly negative while attention-only correlations change sign across configurations (Appendix E). We therefore use ρ only as a descriptive indicator of pressure; residual-chain depth and model-specific geometry are additional factors. At 3 bpp ($\rho \approx 0.07$ for the representative matrix), the pre-PV WikiText-2 gap is 0.65 and shrinks to 0.12 after PV-tuning; OA-EM wins 4 of 6 downstream accuracy tasks and ties one, with a +3.5-point ARC-Easy gain (Appendix B.1). At the standard 2 bpp configuration, the pre-PV gap is 43.2 points and shrinks to 0.23 after PV-tuning. The same persistence extends to Qwen 2.5 14B: OA-EM reduces WikiText-2 perplexity from 9.79 to 8.84 before PV-tuning and from 7.71 to 7.56 after matched PV-tuning, with post-PV C4 improving from 12.73 to 12.62 (Appendix C.4). These results show that PV-tuning substantially reduces, but does not fully erase, matched differences associated with initial codebook geometry.

		Pre-PV		Post-PV	
Model	Rate	Greedy	OA-EM	Greedy	OA-EM
Llama 3B	3 bpp	9.52	**8.87**	8.66	**8.54**
Llama 3B	2 bpp	60.61	**17.39**	11.76	**11.53**
Llama 8B	2 bpp	18.86	**16.38**	9.39	**9.25**
Qwen 3B	2 bpp	12.50	**12.30**	10.93	**10.73**
Qwen 14B	2 bpp	9.79	**8.84**	7.71	**7.56**

Table 4: WikiText-2 perplexity across rates and models (b=8, e=100). Per-model breakdowns in Appendices B and C.

Variation across scales. The 8B Greedy baseline (18.86) is much better than the 3B baseline (60.61) despite a higher representative ρ, reinforcing that ρ is not sufficient to predict catastrophic failure. Model-specific weight and activation geometry are additional factors; we do not claim a unique causal explanation.

We keep K=256 to match AQLM's byte-aligned 8-bit indices; increasing K changes the bitrate as well as ρ (Appendix E).

Cross-dataset variation is reported in Appendix H. Because the calibration domain is not varied experimentally, we make no quantitative domain-distance claim in the main text.

7 Conclusion

This work shows that additive quantization can remain strongly dependent on codebook initialisation even after assignment search and PV-tuning. Controlled decompositions and per-matrix analyses show that the structural ratio ρ should be interpreted as a descriptive measure of representational pressure, not as a sufficient predictor: sequential residual coupling and model-specific error geometry materially affect the outcome.

Across search budgets and four models from 3B to 14B, OA-EM produces lower matched 2-bpp post-PV perplexity than Greedy initialisation. Curvature diagnostics support better preservation of leading eigendirections in aggregate, with mixed total-output error and a gate-projection exception. The matched seed-0 sweep and OA-EM seed-7 replication support search-budget robustness, while the best beam depends on seed and dataset.

Limitations

We evaluate four models from two model families, from 3B to 14B parameters (Llama 3.2 3B, Llama 3.1 8B, Qwen 2.5 3B, Qwen 2.5 14B). The downstream accuracy signal is clearest on Llama 3.2 3B; on Qwen 2.5 3B, the Greedy baseline slightly leads on average zero-shot accuracy (0.606 vs. 0.603) despite worse perplexity. Our method applies directly to free-form additive quantization with sequential residual codebooks. It does not directly transfer to lattice-based (QuIP#, GLVQ) or trellis-based (QTIP) methods, and we do not provide matched latency comparisons against those systems. Likewise, ρ should be treated as a structural descriptor rather than a universal predictor: matrix-level correlations depend on projection family, while fixed-rate decompositions show that residual-chain structure matters even with near-complete codebook utilisation. Beam-width effects are also seed- and dataset-dependent, so we claim robustness of the matched OA-EM advantage rather than monotonic improvement with search budget. We evaluate on English-only benchmarks.

Ethical Considerations

This work improves the efficiency of existing open-weight LLMs. We do not introduce new training data or models. Compression reduces deployment costs and environmental impact but does not address biases present in the original models. Artefact provenance, licensing, and intended use are discussed in Appendix J.

Acknowledgements

This work was supported by the UKRI AI Centre for Doctoral Training in Speech and Language Technologies (SLT) and their Applications funded by UK Research and Innovation [grant number EP/S023062/1]. For the purpose of open access, the authors have applied a Creative Commons Attribution (CC BY) licence to any Author Accepted Manuscript version arising.

We acknowledge IT Services at The University of Sheffield for the provision of services for High Performance Computing.

References

Artem Babenko and Victor Lempitsky. 2014. Additive quantization for extreme vector compression. In *2014 IEEE Conference on Computer Vision and Pattern Recognition*, pages 931–938.

Yonatan Bisk, Rowan Zellers, Ronan Le Bras, Jianfeng Gao, and Yejin Choi. 2020. PIQA: Reasoning about physical commonsense in natural language. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 34, pages 7432–7439.

Jerry Chee, Yaohui Cai, Volodymyr Kuleshov, and Christopher De Sa. 2023. QuIP: 2-bit quantization of large language models with guarantees. In *Advances in Neural Information Processing Systems*, volume 36, pages 4396–4429.

Peter Clark, Isaac Cowhey, Oren Etzioni, Tushar Khot, Ashish Sabharwal, Carissa Schoenick, and Oyvind Tafjord. 2018. Think you have solved question answering? Try ARC, the AI2 reasoning challenge. *Preprint*, arXiv:1803.05457.

A. P. Dempster, N. M. Laird, and D. B. Rubin. 1977. Maximum likelihood from incomplete data via the EM algorithm. *Journal of the Royal Statistical Society: Series B (Methodological)*, 39(1):1–22.

Tim Dettmers, Mike Lewis, Younes Belkada, and Luke Zettlemoyer. 2022. LLM.int8(): 8-bit matrix multiplication for transformers at scale. In *Advances in Neural Information Processing Systems*, volume 35, pages 30318–30332.

Tim Dettmers, Artidoro Pagnoni, Ari Holtzman, and Luke Zettlemoyer. 2023. QLoRA: Efficient finetuning of quantized LLMs. In *Advances in Neural Information Processing Systems*, volume 36, pages 10088–10115.

Tim Dettmers, Ruslan Svirschevski, Vage Egiazarian, Denis Kuznedelev, Elias Frantar, Saleh Ashkboos, Alexander Borzunov, Torsten Hoefler, and Dan Alistarh. 2024. SpQR: A sparse-quantized representation for near-lossless LLM weight compression. In *International Conference on Learning Representations*, pages 5733–5761.

Vage Egiazarian, Andrei Panferov, Denis Kuznedelev, Elias Frantar, Artem Babenko, and Dan Alistarh. 2024. Extreme compression of large language models via additive quantization. In *Proceedings of the 41st International Conference on Machine Learning*, volume 235 of *Proceedings of Machine Learning Research*, pages 12284–12303. PMLR.

Elias Frantar, Saleh Ashkboos, Torsten Hoefler, and Dan Alistarh. 2023. OPTQ: Accurate quantization for generative pre-trained transformers. In *International Conference on Learning Representations*. Also released as GPTQ, arXiv:2210.17323.

Amir Gholami, Sehoon Kim, Zhen Dong, Zhewei Yao, Michael W. Mahoney, and Kurt Keutzer. 2021. A survey of quantization methods for efficient neural network inference. *Preprint*, arXiv:2103.13630.

Aaron Grattafiori, Abhimanyu Dubey, Abhinav Jauhri, Abhinav Pandey, Abhishek Kadian, Ahmad Al-Dahle, Aiesha Letman, Akhil Mathur, Alan Schelten, Alex Vaughan, Amy Yang, Angela Fan, Anirudh Goyal, Anthony Hartshorn, Aobo Yang, Archi Mitra, Archie Sravankumar, Artem Korenev, Arthur Hinsvark, and 542 others. 2024. The Llama 3 Herd of Models. *Preprint*, arXiv:2407.21783.

Hervé Jégou, Matthijs Douze, and Cordelia Schmid. 2011. Product quantization for nearest neighbor search. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 33(1):117–128.

Sehoon Kim, Coleman Richard Charles Hooper, Amir Gholami, Zhen Dong, Xiuyu Li, Sheng Shen, Michael W. Mahoney, and Kurt Keutzer. 2024. SqueezeLLM: Dense-and-sparse quantization. In *Proceedings of the 41st International Conference on Machine Learning*, volume 235 of *Proceedings of Machine Learning Research*, pages 23901–23923. PMLR.

Ji Lin, Jiaming Tang, Haotian Tang, Shang Yang, Wei-Ming Chen, Wei-Chen Wang, Guangxuan Xiao, Xingyu Dang, Chuang Gan, and Song Han. 2024. AWQ: Activation-aware weight quantization for on-device LLM compression and acceleration. In *Proceedings of Machine Learning and Systems*, volume 6, pages 87–100.

Zechun Liu, Changsheng Zhao, Igor Fedorov, Bilge Soran, Dhruv Choudhary, Raghuraman Krishnamoorthi, Vikas Chandra, Yuandong Tian, and Tijmen Blankevoort. 2025. SpinQuant: LLM quantization with learned rotations. In *International Conference on Learning Representations*, volume 2025, pages 92009–92032.

Vladimir Malinovskii, Denis Mazur, Ivan Ilin, Denis Kuznedelev, Konstantin Burlachenko, Kai Yi, Dan Alistarh, and Peter Richtárik. 2024. PV-Tuning: Beyond straight-through estimation for extreme LLM compression. In *Advances in Neural Information Processing Systems*, volume 37, pages 5074–5121.

Vladimir Malinovskii, Andrei Panferov, Ivan Ilin, Han Guo, Peter Richtárik, and Dan Alistarh. 2025. HIGGS: Pushing the limits of large language model quantization via the linearity theorem. In *Proceedings of the 2025 Conference of the Nations of the Americas Chapter of the Association for Computational Linguistics: Human Language Technologies (Volume 1: Long Papers)*, pages 10857–10886, Albuquerque, New Mexico. Association for Computational Linguistics.

Julieta Martinez, Shobhit Zakhmi, Holger H. Hoos, and James J. Little. 2018. LSQ++: Lower running time and higher recall in multi-codebook quantization. In *Computer Vision – ECCV 2018*, pages 508–523, Cham. Springer International Publishing.

Stephen Merity, Caiming Xiong, James Bradbury, and Richard Socher. 2016. Pointer sentinel mixture models. *Preprint*, arXiv:1609.07843.

Denis Paperno, Germán Kruszewski, Angeliki Lazaridou, Ngoc Quan Pham, Raffaella Bernardi, Sandro Pezzelle, Marco Baroni, Gemma Boleda, and Raquel Fernández. 2016. The LAMBADA dataset: Word prediction requiring a broad discourse context. In *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 1525–1534, Berlin, Germany. Association for Computational Linguistics.

Colin Raffel, Noam Shazeer, Adam Roberts, Katherine Lee, Sharan Narang, Michael Matena, Yanqi Zhou, Wei Li, and Peter J. Liu. 2020. Exploring the limits of transfer learning with a unified text-to-text transformer. *Journal of Machine Learning Research*, 21(140):1–67.

Keisuke Sakaguchi, Ronan Le Bras, Chandra Bhagavatula, and Yejin Choi. 2021. WinoGrande: An adversarial Winograd schema challenge at scale. *Communications of the ACM*, 64(9):99–106.

Wenqi Shao, Mengzhao Chen, Zhaoyang Zhang, Peng Xu, Lirui Zhao, Zhiqian Li, Kaipeng Zhang, Peng Gao, Yu Qiao, and Ping Luo. 2024. OmniQuant: Omnidirectionally calibrated quantization for large language models. In *International Conference on Learning Representations*, volume 2024, pages 45472–45496.

Lintang Sutawika, Hailey Schoelkopf, Leo Gao, Baber Abbasi, Stella Biderman, Jonathan Tow, Ben Fattori, Charles Lovering, farzanehnakhaee70, Jason Phang, Anish Thite, Fazz, Aflah, Niklas Muennighoff, Thomas Wang, sdtblck, nopperl, gakada, tttyuntian, and 11 others. 2024. EleutherAI/lm-evaluation-harness: v0.4.3.

Albert Tseng, Jerry Chee, Qingyao Sun, Volodymyr Kuleshov, and Christopher De Sa. 2024a. QuIP#: Even better LLM quantization with Hadamard incoherence and lattice codebooks. In *Proceedings of the 41st International Conference on Machine Learning*, volume 235 of *Proceedings of Machine Learning Research*, pages 48630–48656. PMLR.

Albert Tseng, Qingyao Sun, David Hou, and Christopher De Sa. 2024b. QTIP: Quantization with trellises and incoherence processing. In *Advances in Neural Information Processing Systems*, volume 37, pages 59597–59620.

Mart van Baalen, Andrey Kuzmin, Ivan Koryakovskiy, Markus Nagel, Peter Couperus, Cedric Bastoul, Eric Mahurin, Tijmen Blankevoort, and Paul Whatmough. 2025. GPTVQ: The blessing of dimensionality for LLM quantization. *Preprint*, arXiv:2402.15319.

Guangxuan Xiao, Ji Lin, Mickael Seznec, Hao Wu, Julien Demouth, and Song Han. 2023. SmoothQuant: Accurate and efficient post-training quantization for large language models. In *Proceedings of the 40th International Conference on Machine Learning*, volume 202 of *Proceedings of Machine Learning Research*, pages 38087–38099. PMLR.

An Yang, Baosong Yang, Beichen Zhang, Binyuan Hui, Bo Zheng, Bowen Yu, Chengyuan Li, Dayiheng Liu, Fei Huang, Haoran Wei, Huan Lin, Jian Yang, Jianhong Tu, Jianwei Zhang, Jianxin Yang, Jiaxi Yang, Jingren Zhou, Junyang Lin, Kai Dang, and 23 others. 2025. Qwen2.5 technical report. *Preprint*, arXiv:2412.15115.

Rowan Zellers, Ari Holtzman, Yonatan Bisk, Ali Farhadi, and Yejin Choi. 2019. HellaSwag: Can a machine really finish your sentence? In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 4791–4800, Florence, Italy. Association for Computational Linguistics.

Xi Zhang, Xiaolin Wu, Jiamang Wang, and Weisi Lin. 2025. Learning grouped lattice vector quantizers for low-bit LLM compression. In *Advances in Neural Information Processing Systems*, volume 38, pages 122657–122683.

A Pareto Analysis

Table 5 combines post-PV WikiText-2 perplexity, mean zero-shot accuracy, and quantization time for each matched configuration. At every full-search beam width (e=100), OA-EM attains lower perplexity and higher accuracy with equal or lower

quantization time; the short-budget setting (e=5) is the only case where OA-EM is slower.

Init	Config	Wiki-2↓	Avg↑	Q-time
Greedy	b=4, e=100	12.66	.573	6.1h
OA-EM	b=4, e=100	**11.53**	**.589**	6.1h
Greedy	b=8, e=100	11.76	.585	9.9h
OA-EM	b=8, e=100	**11.53**	**.591**	**9.2h**
Greedy	b=16, e=100	12.01	.584	16.9h
OA-EM	b=16, e=100	**11.49**	**.591**	**15.5h**
Greedy	b=8, e=5	12.69	.563	**6.8h**
OA-EM	b=8, e=5	**11.76**	**.572**	7.3h

Table 5: Quality–compute Pareto analysis at 2 bpp on Llama 3.2 3B. All metrics post-PV-tuning.

B Overcomplete Regime: 3 bpp Detailed Results

Table 6 presents the pre-PV-tuning perplexity at 3 bpp on Llama 3.2 3B.

	Wiki-2↓	C4↓
FP16	7.28	11.04
Greedy	9.52	**13.39**
OA-EM	**8.87**	13.51

Table 6: Pre-PV-tuning perplexity at 3 bpp on Llama 3.2 3B (b=8, e=100).

B.1 3 bpp Post-PV-Tuning Downstream

After PV-tuning, the gap at 3 bpp shrinks from 0.65 to 0.12 (WikiText-2: 8.66 vs. 8.54; C4: 11.43 vs. 11.45). Table 7 reports the full post-PV downstream evaluation. OA-EM wins 4 of 6 accuracy tasks and ties on 1, with the largest gains on ARC-Easy (+3.5pp) and LAMBADA accuracy (+1.5pp). It also achieves the lowest LAMBADA perplexity. Average accuracy increases by 0.7 percentage points (0.647 → 0.654), with the only notable regression on WinoGrande (−1.2pp). These results indicate that basin persistence extends to the overcomplete regime.

Task	Greedy	OA-EM
ARC-C	.418	**.421**
ARC-E	.658	**.693**
HellaSwag	.704	.704
PIQA	.760	**.762**
WinoGrande	**.669**	.657
LAMB. acc	.670	**.685**
LAMB. ppl↓	4.87	**4.64**
Avg	.647	**.654**

Table 7: Post-PV-tuning downstream at 3 bpp on Llama 3.2 3B (b=8, e=100). Avg = mean of the six accuracy tasks.

C Full Downstream and Perplexity Results

This appendix provides the complete per-task downstream evaluations and per-model perplexity breakdowns that underlie the summary in Table 8 and Table 4. Results are organised by model.

		Avg Acc↑	
Model	Config	Greedy	OA-EM
Llama 3B	b=4	.573	**.589**
Llama 3B	b=8	.585	**.591**
Llama 3B	b=16	.584	**.591**
Llama 3B	b=8, e=5	.563	**.572**
Llama 8B	b=8	.649	**.656**
Qwen 3B	b=8	**.606**	.603

Table 8: Post-PV-tuning downstream summary at 2 bpp. Avg = mean of 6 zero-shot accuracy tasks. Per-task results in Tables 9 and 11.

C.1 Llama 3.2 3B at 2 bpp

Table 9 presents post-PV-tuning downstream results across all four beam configurations. OA-EM wins or ties average accuracy in every setting; the advantage is largest at b=4 (+1.6pp), where the greedy baseline has minimal search to compensate for poor initialisation.

	b=4		b=8		b=16		e=5	
Task	G	OA	G	OA	G	OA	G	OA
ARC-C	.350	**.359**	.356	**.366**	**.363**	.357	**.332**	.331
ARC-E	.560	**.614**	.600	**.601**	.619	**.624**	**.569**	.561
HellaSwag	.620	**.625**	.623	**.626**	.625	**.627**	.611	**.625**
PIQA	.734	.734	**.741**	.736	**.731**	.726	.716	**.734**
WinoGrande	.594	**.619**	**.614**	.611	.589	**.624**	.594	**.609**
LAMB. acc	.577	**.584**	.573	**.604**	.579	**.587**	.556	**.574**
LAMB. ppl↓	7.65	**7.13**	7.96	**6.95**	**7.52**	7.66	8.61	**7.73**
Avg	.573	**.589**	.585	**.591**	.584	**.591**	.563	**.572**

Table 9: Post-PV downstream accuracy, Llama 3.2 3B, 2 bpp, across beam configurations (e=100 unless noted; e=5 is the short budget at b=8). G = Greedy, OA = OA-EM. Avg = mean of the six accuracy tasks (LAMBADA perplexity excluded).

C.2 Llama 3.1 8B at 2 bpp

Table 10 shows the perplexity breakdown before and after PV-tuning. Table 11 presents post-PV-tuning downstream results. Before PV-tuning, Greedy/OA-EM reach WikiText-2 perplexities of 18.86/16.38 and C4 of 15.01/14.50.

	Before PV		After PV	
Init	Wiki-2	C4	Wiki-2	C4
Llama 3.1 8B				
Greedy	18.86	15.01	9.39	12.02
OA-EM	**16.38**	**14.50**	**9.25**	**11.89**
Qwen 2.5 3B				
Greedy	12.50	**16.01**	10.93	14.57
OA-EM	**12.30**	16.08	**10.73**	**14.49**

Table 10: WikiText-2 / C4 perplexity before and after PV-tuning at 2 bpp (b=8, e=100) for Llama 3.1 8B and Qwen 2.5 3B.

	Llama 3.1 8B		Qwen 2.5 3B	
Task	G	OA	G	OA
ARC-C	**.432**	.424	**.375**	.366
ARC-E	.656	**.677**	**.662**	.651
HellaSwag	.707	**.714**	.626	**.630**
PIQA	.759	**.769**	**.739**	.737
WinoGrande	.661	**.679**	.634	**.647**
LAMB. acc	**.679**	.675	**.600**	.587
LAMB. ppl↓	4.60	**4.59**	**7.29**	7.37
Avg	.649	**.656**	**.606**	.603

Table 11: Post-PV downstream accuracy at 2 bpp (b=8, e=100) for Llama 3.1 8B and Qwen 2.5 3B. G = Greedy, OA = OA-EM.

C.3 Qwen 2.5 3B at 2 bpp

Table 10 presents the perplexity breakdown; OA-EM wins on both metrics after PV-tuning despite the mild bottleneck on this architecture. Table 11 shows the downstream evaluation where the baseline nominally leads on average accuracy (0.606 vs. 0.603), as noted in the main text.

C.4 Qwen 2.5 14B at 2 bpp

We quantize Qwen 2.5 14B at 2 bpp with K=256, M=2, g=8, b=8, e=100, seed 42, and matched C4 calibration data. The Greedy and OA-EM runs use matched quantization and PV configurations, extending the 3B/8B evaluation to 14B.

	Before PV		After PV	
Init	Wiki-2	C4	Wiki-2	C4
Greedy	9.79	**13.52**	7.71	12.73
OA-EM	**8.84**	13.54	**7.56**	**12.62**

Table 12: WikiText-2 / C4 perplexity before and after matched PV-tuning at 2 bpp (b=8, e=100) for Qwen 2.5 14B.

Before PV, OA-EM improves WikiText-2 perplexity by 9.70%; C4 is slightly worse (13.52 vs. 13.54). Mean calibration layer-output error decreases from 9.23×10^{-4} to 8.55×10^{-4} (7.37% lower), and median error from 4.23×10^{-4} to 4.06×10^{-4} (4.02% lower), with improvements in all 48 layers.

PV data and checkpoint selection. The matched PV configuration uses seed 42, batch 32, learning rate 3×10^{-4}, and a five-epoch budget. The data comprise 10,000 C4 samples. Checkpoints minimise WikiText-2 *test* perplexity; C4 validation perplexity is reported from the same checkpoint, not selected separately.

After matched PV-tuning, OA-EM reduces WikiText-2 perplexity from 7.71 to 7.56 (1.95%) and paired C4 perplexity from 12.73 to 12.62 (0.86%). Thus the initialisation advantage extends to 14B and is not fully erased by end-to-end PV refinement.

Compute. On the same B200 host, Greedy quantization took 35h 27m and OA-EM took 31h 37m. The Qwen 2.5 14B analysis covers perplexity and layer-output reconstruction.

D Controlled Mechanism Ablations

The main text reports only the headline-controlled results; this appendix provides the full fixed-bitrate decompositions, the Euclidean-EM ablation, and the curvature diagnostic.

D.1 Controlled Fixed-Bitrate Decompositions

Table 13 holds the bitrate at 2 bpp while changing the residual decomposition. With K=256, increasing the group size gives $M \in \{1, 2, 4\}$ and changes K^M by orders of magnitude. Greedy is suboptimal but not catastrophic for M=1 (21.39 vs. 15.66), fails most severely for the standard M=2 configuration (60.61 vs. 17.39), and improves again in the highly overcomplete M=4 setting (19.08 vs. 16.94). Thus high ρ creates representational pressure but does not by itself imply catastrophic failure: the absence of a residual chain at M=1 removes the sequential coupling mechanism.

g	K	M	K^M	Greedy	OA-EM
4	256	1	256	21.39	**15.66**
8	256	2	65,536	60.61	**17.39**
16	256	4	$\approx 4.3 \times 10^9$	19.08	**16.94**

Table 13: Controlled 2-bpp sweep on Llama 3.2 3B. WikiText-2 perplexity before PV-tuning. Bitrate is held fixed while residual-chain depth and representational capacity change.

A complementary decomposition holds g=8, the nominal index rate, and the nominal tuple count K^M=65,536 fixed. Table 14 makes explicit that (K=256, M=2) and (K=16, M=4) nevertheless differ in additive geometry, residual-chain depth, and learned codebook parameter count. Thus this experiment does not isolate chain depth independently of all other structural changes.

K	M	Index bpp	K^M	MKg
256	2	2.00	65,536	4,096
16	4	2.00	65,536	512

Table 14: Equal nominal index rate and tuple count at $g = 8$. MKg is the number of scalar codebook parameters per codebook set.

For $(16, 4)$, OA-EM reduces pre-PV WikiText-2 perplexity by 90.44% and C4 by 37.58%; the post-PV reductions are 16.5% and 4.7% (Table 15). Both methods are worse than their standard $(256, 2)$ counterparts despite near-complete codebook utilisation (Table 20). Equal K^M, and hence equal ρ, is therefore insufficient to predict performance. This supports considering the decomposition as well as structural pressure, without attributing the entire difference to chain depth alone.

	Pre-PV		Post-PV	
Method	Wiki-2	C4	Wiki-2	C4
Greedy	274.93	49.73	17.16	18.00
OA-EM	**26.28**	**31.04**	**14.33**	**17.15**

Table 15: WikiText-2 and C4 perplexity for Llama 3.2 3B with $(K, M, g) = (16, 4, 8)$, before and after PV-tuning (seed 42, b=8, e=100).

D.2 EM versus Output-Aware EM

To separate generic iterative refinement from Hessian/output awareness, Table 16 replaces the OA-EM metric with Euclidean distance while keeping the EM refinement structure. Euclidean EM improves pre-PV WikiText-2 over Greedy (60.61 $\rightarrow$ 43.41) but remains far behind OA-EM (17.39), and its C4 and post-PV results are worse than both OA-EM and Greedy. This isolates a benefit from output-aware curvature weighting beyond EM refinement alone.

	Pre-PV		Post-PV	
Init	Wiki-2	C4	Wiki-2	C4
Greedy	60.61	18.64	11.76	14.30
Euclidean EM	43.41	23.28	12.20	16.33
OA-EM	**17.39**	**18.00**	**11.53**	**14.26**

Table 16: Euclidean-EM ablation on Llama 3.2 3B at 2 bpp (K=256, M=2, g=8, b=8).

D.3 Curvature Diagnostic

We evaluate pre-PV seed-42 Greedy, Euclidean EM, and OA-EM checkpoints of Llama 3.2 3B on 32 held-out C4 sequences. All methods use the same reference activations, per-matrix Hessian, and eigenspaces. The reference model is BF16; diagnostic weights, covariance matrices, norms, and eigendecompositions are cast or accumulated in FP32. All seven projections in all 28 layers are included: 196 matrices per method and 588 method–matrix measurements.

Table 17 reports ordinary weight NMSE, total held-out output NMSE, and curvature NMSE restricted to the leading $k \in \{1, 8, 32, 128\}$ Hessian eigendirections. Ratios are computed within each matrix against Greedy and then summarised by their median; a win means a ratio strictly below one. These are descriptive matrix-level comparisons, not independent experimental replications.

Metric	Euclid. EM / Greedy Ratio	Euclid. EM / Greedy Wins	OA-EM / Greedy Ratio	OA-EM / Greedy Wins
Weight NMSE	**0.999**	**114/196**	1.045	5/196
Output NMSE	**0.998**	**127/196**	1.002	95/196
Curvature NMSE				
Top-1	0.981	119/196	**0.887**	**146/196**
Top-8	0.984	128/196	**0.874**	**160/196**
Top-32	0.987	130/196	**0.871**	**168/196**
Top-128	0.991	132/196	**0.888**	**170/196**

Table 17: Held-out mechanism analysis over all 196 matrices of Llama 3.2 3B. Ratios are medians of per-matrix ratios against Greedy; lower is better and the Greedy reference is 1. Output NMSE is total held-out output NMSE.

OA-EM increases median weight NMSE by 4.55% but reduces median leading-eigenspace curvature NMSE by 11.2–12.9%, with 146–170 wins. Total output NMSE is mixed: the median ratio is 1.002 (0.25% worse), the mean ratio is 0.987, and OA-EM wins 95/196 matrices. Thus the evidence supports preferential preservation of high-curvature directions in aggregate, not uniformly better total-output reconstruction.

Table 18 shows that the improvement is projection-dependent. Attention projections improve by 16.9–19.1%, with 26–28 wins out of 28. MLP up/down projections also improve, whereas `gate_proj` worsens by 7.7% and wins only 7/28 matrices. This boundary condition qualifies the aggregate mechanism claim.

Projection	Ratio	Wins
`q_proj`	0.814	27/28
`k_proj`	0.831	26/28
`v_proj`	0.829	27/28
`o_proj`	0.809	28/28
`up_proj`	0.935	27/28
`down_proj`	0.921	26/28
`gate_proj`	1.077	7/28

Table 18: Top-32 curvature NMSE by projection. Ratio is the median OA-EM/Greedy ratio.

E Representational-Ratio Details

For matrix ℓ, we compute $N_\ell = \#\text{weights}_\ell/g$ and $\rho_\ell = N_\ell/K^M$. These quantities are unchanged by initialisation or PV-tuning. Code assignments can change during PV-tuning, which enables code updates; assignment utilisation is therefore distinct from structural ρ.

Matrix-level correlation. Let E_ℓ^{G} and E_ℓ^{OA} be the logged reconstruction errors for matched Greedy and OA-EM matrices. We correlate $\log_{10}\rho_\ell$ with the relative reduction $1 - E_\ell^{\mathrm{OA}}/E_\ell^{\mathrm{G}}$, not with downstream perplexity. Table 19 gives Pearson coefficients and matched matrix counts.

Model	(K, M)	Pairs	r (all proj.)
Llama 3.2 3B	$(16, 4)$	196	-0.270
Llama 3.2 3B	$(256, 2)$	196	-0.170
Llama 3.1 8B	$(256, 2)$	224	-0.233
Qwen 2.5 14B	$(256, 2)$	336	-0.129

Table 19: Pearson correlation r of $\log_{10}\rho$ with relative OA-EM reconstruction-error reduction over matched matrix pairs.

All overall coefficients are negative. The attention-only coefficients are not stable across configurations and change sign between Llama 3B $(16, 4)$ and Llama 8B $(256, 2)$. These are descriptive relationships: matrices share layers and model parameters, and matrix dimensions and projection type confound the association. They do not establish a universal predictive or causal relationship between ρ and the OA-EM gain.

Codebook utilisation. For marginal assignment probabilities p, active fraction measures the proportion of codewords used, normalised entropy is $H(p)/\log K$, and effective codewords is $\exp H(p)$ (natural-log entropy). Table 20 averages the marginal statistics over codebooks within each matrix and then over matrices. Unique tuples counts the distinct joint index combinations per matrix. All measurements are pre-PV.

Init	Norm. entropy	Eff. codewords	Unique tuples
Llama 3B, (K, M)=$(16, 4)$			
Greedy	0.9999	16.00	64,986
OA-EM	0.9995	15.98	64,442
Llama 3B, (K, M)=$(256, 2)$			
Greedy	0.9982	253.43	65,201
OA-EM	0.9970	251.77	64,939
Llama 8B, (K, M)=$(256, 2)$			
Greedy	0.9981	253.30	65,315
OA-EM	0.9963	250.90	65,012
Qwen 14B, (K, M)=$(256, 2)$			
Greedy	0.9970	252.01	65,430
OA-EM	0.9943	248.45	65,107

Table 20: Mean pre-PV code usage. The active fraction is 1.0000 (every codeword used) in all configurations.

All marginal codebooks are fully active with near-maximal entropy. Even the poorly performing $(K{=}16, M{=}4)$ configuration uses nearly all nominal tuples on average. Its failure is therefore

not explained by simple codeword collapse. Equal tuple count does not imply equal additive geometry or equal learned parameter count (Table 14).

Codebook-size trade-off. The main experiments follow AQLM's 8-bit codebook indices. Increasing K reduces ρ but also changes the operating point: for $M{=}2, g{=}8$, $K{=}512$ requires 9-bit indices and raises the nominal index rate from 2.0 to 2.25 bits per weight, while breaking the byte-aligned indexing convention. For this reason the $K{=}512$ experiment was not run; a systematic K–bitrate trade-off is left to future work.

F Controlled Beam–Seed Robustness

Table 21 separates three sources of evidence: a fully matched Greedy/OA-EM seed-0 sweep; an OA-EM-only seed-7 replication; and the original seed-42 quantizations.

OA-EM improves both perplexity measures at every beam before and after PV in the fully matched seed-0 sweep. Its WikiText-2 curve is non-monotonic: beam 8 is best before and after PV, while seed 7 favours beam 16. On seed 0, PV reduces the OA-EM WikiText-2 best-to-worst range from 1.75 to 0.34. C4 can favour a different beam. These results support robustness over the tested search budgets, not universal beam monotonicity.

		Pre-PV		Post-PV	
Beam	Init	Wiki-2	C4	Wiki-2	C4
Seed 0 (matched sweep)					
4	Greedy	126.15	25.14	12.76	16.44
4	OA-EM	**17.85**	**22.36**	**12.27**	**16.23**
8	Greedy	96.45	25.08	12.93	16.50
8	OA-EM	**16.81**	**22.18**	**12.23**	**16.27**
16	Greedy	82.46	26.94	12.91	16.46
16	OA-EM	**18.56**	**22.09**	**12.57**	**15.81**
Seed 7 (OA-EM replication)					
4	OA-EM	17.67	21.90	12.13	16.21
8	OA-EM	17.55	21.56	12.06	16.13
16	OA-EM	17.20	21.68	12.05	16.12
Seed 42 (original runs)					
4	Greedy	352.39	20.66	12.66	14.61
4	OA-EM	**16.82**	**18.09**	**11.53**	**14.30**
8	Greedy	60.61	18.64	11.76	14.30
8	OA-EM	**17.39**	**18.00**	**11.53**	**14.26**
16	Greedy	46.01	19.00	12.01	14.25
16	OA-EM	**16.53**	**17.92**	**11.49**	**14.22**

Table 21: Beam–seed robustness for Llama 3.2 3B at 2 bpp ($K{=}256$, $M{=}2$, $g{=}8$, $e{=}100$). Seed 0 is a matched Greedy/OA-EM sweep, seed 7 an OA-EM-only replication, and seed 42 the original pre-PV runs with standardised post-PV results.

G End-to-End Wall-Clock Accounting

For Llama 3.2 3B, PV-tuning takes approximately 2.0h on a single A100 and uses the same configuration for both initialisations. Adding this common cost to Table 3 gives approximately 8.1h end-to-end for OA-EM at $b{=}4$, 11.2h at $b{=}8$, and 17.5h at $b{=}16$; the corresponding Greedy totals are 8.1h, 11.9h, and 18.9h.

H Cross-Dataset Variation

Table 22 compares the Greedy/OA-EM perplexity ratio on the calibration domain (C4) and on an external evaluation set (WikiText-2). The calibration domain is not varied, so we do not draw a quantitative domain-distance conclusion.

Benchmark	Role	Greedy	OA-EM	Ratio
C4	Calibration domain	18.64	**18.00**	1.04×
WikiText-2	External eval.	60.61	**17.39**	3.49×

Table 22: Perplexity across evaluation datasets at 2 bpp on Llama 3.2 3B ($b{=}8$, $e{=}100$, before PV-tuning). Ratio = Greedy/OA-EM.

I Proof of Proposition 1

Proof. Let $\mathbf{r}^g = \mathbf{w} - \mathbf{c}_{1,i^g} = \mathbf{r}^* - \boldsymbol{\delta}$. Then:

$$\begin{aligned}\varepsilon^g &= \|\mathbf{r}^g - \mathbf{c}_{2,j^g}\|^2 = \|\mathbf{r}^* - \boldsymbol{\delta} - \mathbf{c}_{2,j^g}\|^2 \\ &= \|\mathbf{r}^* - \mathbf{c}_{2,j^g}\|^2 + \|\boldsymbol{\delta}\|^2 - 2\langle \mathbf{r}^* - \mathbf{c}_{2,j^g},\, \boldsymbol{\delta}\rangle \end{aligned} \tag{5}$$

Subtracting $\varepsilon^* = \|\mathbf{r}^* - \mathbf{c}_{2,j^*}\|^2$ and rearranging yields Eq. (2). The residual mismatch $\|\mathbf{r}^* - \mathbf{c}_{2,j^g}\|^2 - \|\mathbf{r}^* - \mathbf{c}_{2,j^*}\|^2 \geq 0$ since j^* minimises over $\mathbf{r}^*$. □

J Scientific Artefacts and Responsible Use

Artefact provenance. Our implementation extends AQLM (Egiazarian et al., 2024) with OA-EM initialisation and uses PV-tuning (Malinovskii et al., 2024) for subsequent refinement. The implementation is available at `https://github.com/kenno94-IK/aqlm-oaem`. We evaluate the pretrained Llama 3.2 3B and Llama 3.1 8B models (Grattafiori et al., 2024) and Qwen 2.5 3B and Qwen 2.5 14B models (Yang et al., 2025). Downstream evaluation uses the LM Evaluation Harness (Sutawika et al., 2024).

Software and model licences. The upstream AQLM implementation is distributed under the

Apache License 2.0, and our OA-EM implementation is distributed under the Apache License 2.0. The LM Evaluation Harness is distributed under the MIT License. Llama 3.2 3B and Llama 3.1 8B are subject to the Llama 3.2 and Llama 3.1 Community License Agreements, respectively, including their associated acceptable-use conditions. Qwen 2.5 3B is subject to the Qwen Research License Agreement, which limits use to non-commercial research and evaluation unless separate commercial permission is obtained. Qwen 2.5 14B is distributed under the Apache License 2.0. Any redistribution of quantized model derivatives remains subject to the applicable base-model terms; the licence for our implementation does not replace those terms.

Dataset licences. The dataset providers identify C4 as ODC-By, ARC as CC BY-SA 4.0, HellaSwag as MIT, PIQA as Academic Free License 3.0, and LAMBADA as CC BY 4.0. The WinoGrande authors identify the dataset licence as CC BY, separately from the Apache License 2.0 covering their codebase. WikiText is distributed under Creative Commons Attribution-ShareAlike terms; its hosted dataset card contains inconsistent version information, listing CC BY-SA 3.0 and GFDL in its metadata but referring to CC BY-SA 4.0 in its licensing description.[1]

Intended use. We use the pretrained models and existing datasets for research on model compression, reconstruction error, language modelling, and zero-shot benchmark evaluation. These uses follow the modelling and evaluation purposes for which the artefacts are provided. The intended use of our implementation is to reproduce, analyse, and extend the OA-EM quantization experiments. Use of the implementation with a particular model or dataset must respect that artefact's access conditions and licence, including the research restrictions applying to Qwen 2.5 3B.

[1] https://huggingface.co/datasets/Salesforce/wikitext